A New CAPTCHA Interface Design for Mobile Devices

Rosa Lin[1] **Shih-Yu Huang**[1] **Graeme B Bell**[2] **Yeuan-Kuen Lee**[1]

[1]Dept. of CSIE, Ming Chuan University, 5 Teh-Ming Rd, Guei-Shan, Taoyuan, Taiwan 333.
[2]School of IT, Murdoch University, WA, Australia.
Email: syhuang@mail.mcu.edu.tw

Abstract

This paper discusses and demonstrates the interplay between system security and user interface convenience in CAPTCHA design, and in particular, mobile device CAPTCHA design. A CAPTCHA is a computer-based security test used to distinguish human users from artificial users, preventing automated abuse of networked resources. As mobile network services improve, we can anticipate that future mobile network services will come under attack from automated programs. Importantly, while CAPTCHA techniques have existed for Internet services for some time, only limited work has been carried out to establish CAPTCHAs suitable for mobile device interfaces. The Drawing CAPTCHA (2006) is one of the most well known systems of this type. Unfortunately, though it is straightforward, it is not secure. To demonstrate this, an image-processing technique is newly proposed that breaks the Drawing CAPTCHA. A new CAPTCHA approach is then introduced here which is intended specifically for mobile devices. Experimental results suggest that this new CAPTCHA design is user-friendly as well as secure.

Keywords: CAPTCHA, Mobile devices, Security UI

1 Introduction

In recent years, the development of the information technology sector has meant that peoples day-to-day use of the Internet has continually increased; and the convenience of Internet services has increased in kind. Unfortunately, abusive users such as hackers can exploit these internet resources for their own purposes by using automated bots (simulated users) that can reduce the performance of online systems for legitimate users. In order to avoid this situation, it is important to be able to distinguish between valid human users and invalid computer bots on the Internet. The Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA) was created to meet this need. A CAPTCHA is a form of Turing Test that distinguishes human users and computer bots automatically (Pope & Kaur 2005). The test is designed so that human users can answer any questions or challenges easily, but computer-program based imitators face considerably greater difficulty. By considering the quality or correctness of a response, a judging computer can determine whether the tested user is a human or a computer bot.

Figure 1: Example of a typical reading-based CAPTCHA system.

This makes CAPTCHAs an interesting form of user interface in two senses. Firstly, the CAPTCHA interface is the essence of the application; remove the interface and there is nothing left over. Secondly, these interfaces attempt to optimally *inconvenience* one class of users - ideally to the point where recognising or interpreting the interface is entirely intractable; while simultaneously targeting the traditional HCI goal of maximal convenience for regular users.

Internet-based CAPTCHAs are generally of two forms: reading-based CAPTCHAs, and image-based CAPTCHAs. Fig. 1 shows an example of a reading-based CAPTCHA. This kind of CAPTCHA is usually composed of warped English characters and Arabic numerals, overlaid by straight or curved lines that are generated randomly and act as image noise. The challenge is to identify the characters that are obscured by image noise and warping, and enter them into a text box. An automated bot has difficulty deciphering the underlying characters, since until recently there have been no character recognition techniques that can understand what the characters are in the presence of so much noise. On the other hand, humans can answer the question correctly using their natural abilities when faced with the task of character recognition in a noisy environment. Humans use innate skills such as visual continuity and learned knowledge (e.g. familiarity with computer fonts) in order to distinguish letters and numerals by their shapes. In the example shown in Fig. 1, a human user could type the correct answer "BA7WVr8TX" very easily. If a user responds to this kind of test correctly, the system employing the CAPTCHA test to protect its online resources will consider the user to be human. Otherwise, the user is considered to be potentially an automated bot; attempts to use resources will be rejected.

Academic research into CAPTCHAs has the form of a friendly arms race. Typically, one group of researchers act as malicious users that attempt to defeat the latest CAPTCHA systems automatically, e.g. (Mori & Malik 2005, Moy et al. 2004, Chellapilla et al. 2005, Chellapilla & Simard 2005). Meanwhile, an opposing group of researchers try to design new defensive CAPTCHA techniques in response to established or anticipated attacks (Coates et al. 2001, Hoque et al. 2006, Baird & Bentley 2005, Misra & Gaj 2006). An effective CAPTCHA system should balance the needs of both computer security and human-friendliness. Unfortunately, in practice, balancing these two opposing needs is difficult (Huang et al. 2008).

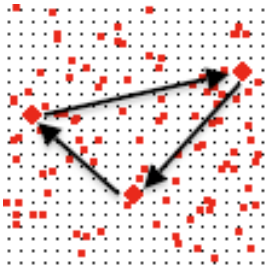


Figure 2: Example of the Drawing CAPTCHA for mobile devices. The user must connect the three red diamonds with lines, to form a triangle. A correct solution is indicated with arrowed black lines.

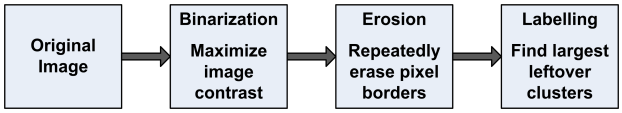


Figure 3: Overview of an algorithm for breaking the Drawing CAPTCHA.

Internet-based CAPTCHAs suitable for home PCs are relatively mature as a technology; but there is still a need to effectively address the problem of designing CAPTCHAs suitable for the hardware interface of common mobile devices. Consequently, this paper proposes a new form of image-based CAPTCHA interface design, well-suited for mobile devices. The design utilizes the convenience of the touch-screen interfaces of mobile devices, and it is intended to be particularly approachable for younger or non-technical mobile device users. The CAPTCHA relies on implicit human knowledge of 3-D world objects and scenes, and the natural ability to recognize both overlapping images and 2-D projections of 3-D objects.

The rest of this paper is organized as follows. Section II introduces the detail of the related Drawing CAPTCHA system. Section III presents the proposed erosion-based CAPTCHA-breaking algorithm which operates against the Drawing CAPTCHA system. Section IV introduces the proposed new "CAPTCHA Zoo" scheme for mobile devices and presents our analysis. Finally, Section V provides our conclusions.

2 Related Work: The Drawing CAPTCHA

The majority of CAPTCHA techniques are designed for use on home computers or laptops. However, increasingly people are accessing network resources using smaller mobile devices as well as regular computers. The development of a CAPTCHA user-verification mechanism suitable for mobile devices is therefore an issue which merits research attention. In particular, the capabilities of touch-screen user-interfaces in mobile devices have proven very convenient for users. People can use their fingers directly on the screen surface to handle all device operations, from dialling voice calls and browsing websites to playing games and manipulating graphics. In 2006, a CAPTCHA mechanism for mobile devices was proposed in (Shirali-Shahreza et al. 2006) which was appropriate for use on a touch-screen device, named the Drawing CAPTCHA. A Drawing CAPTCHA is an image-based CAPTCHA whereby users draw appropriate lines on a screen in order to pass the CAPTCHA challenge.

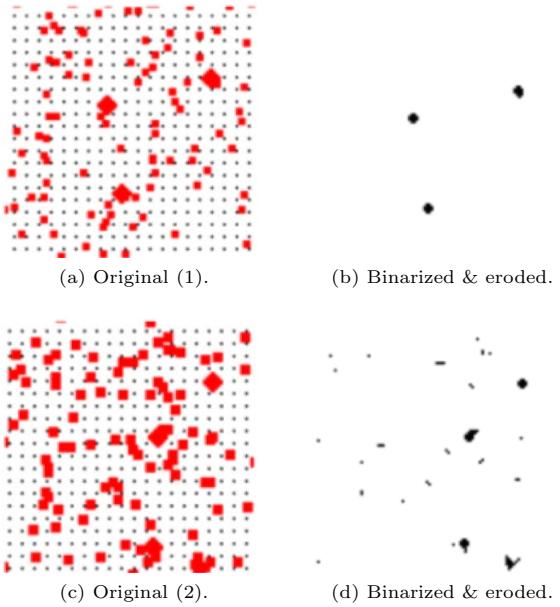
Fig. 2 shows an example of a Drawing CAPTCHA that is intended for use on a mobile phone. This system generates a gray-colored dotted background to begin with as an obfuscation measure against image-processing attacks. A significant number of large square dots and a small number of even larger diamond-shaped dots are then randomly drawn on the screen. The user is asked to connect the diamond-shaped dots. Therefore, the user must first find three diamond-shaped dots; and secondly, connect them to each other by drawing lines. There is no need to connect the dots in a sequence. If a user can respond appropriately, they are considered to be human, but if not, they are considered to be potentially an abusive automated program, and denied access.

This approach has two obvious advantages. Humans are currently considerably better able to recognize visually presented shapes in the presence of noise than computers. Secondly, the interface - drawing lines - is well suited to the touch-screen of many 3G mobile devices. It may be observed that it is rather more convenient for a user to draw lines with their finger or a pointing device, than enter characters using a small virtual keyboard on the screen or using the small set of buttons available as part of the numerical keypad on most mobile devices. Unfortunately, the shapes involved are often too small to allow convenient human operation, particularly on small mobile device screens. In addition, the shapes used (squares and diamonds) provide clues that can allow image-processing techniques to break the CAPTCHA. To demonstrate this problem, this paper now proposes an effective image-processing based erosion algorithm that breaks instances of the Drawing CAPTCHA.

3 Erosion-based Breaking Algorithm for the Drawing CAPTCHA

The relative sizes of the different dots within the Drawing CAPTCHA give us useful clues that allow us to attack it. Since the sizes of background dots and the 'noise/clutter' square dots are smaller than the diamond-shaped dots, they can be filtered out by several erosion operations. Erosion is an established image-processing technique whereby line borders are erased by one pixels depth per iteration. This paper proposes a three-phase CAPTCHA-breaking algorithm that defeats the Drawing CAPTCHA by exploiting the vulnerability of the relative size property to erosion algorithms. Fig. 3 shows the structure of the proposed algorithm. The first phase involves binarization, whereby the original greyscale or colored bitmap image is transformed to a high contrast black and white image. Next, the binarized image is transformed by multiple iterations of an erosion process, which has the effect of erasing both the background dots and also the cluttering square dots. Finally, the labeling phase finds all of the remaining connected components in the image and considers the largest to be candidate diamond-shaped dots. This attack differs from Chellapilla's attacks in three ways. Firstly, the domain - we attack a graphical CAPTCHA rather than a text-based CAPTCHA. Here, it is not so important to preserve fine shape detail. Secondly, the manner of attack. We do not need to engage in enlargement or iterative dilation. Erosion by itself rapidly isolates the diamonds. Finally, our attack lacks OCR. A weakness of the Drawing CAPTCHA is that the targets are simple, identical objects.

Fig. 4 demonstrates the attack against two example CAPTCHAs. Fig. 4a shows an original Drawing CAPTCHA challenge image with three diamond-shaped dots and 100 square dots. Fig. 4b shows the results of the first and second phases of the proposed erosion-based breaking algorithm, when three iterations of erosion have been performed. It is easy to see that the locations of the three largest connected components in Fig. 4b are the locations of



(a) Original (1). (b) Binarized & eroded.

(c) Original (2). (d) Binarized & eroded.

Figure 4: Applying the breaking algorithm to two original Drawing CAPTCHA instances.

the diamond-shaped dots in the original image, and the algorithm operates successfully. However, this is not always the case. Fig. 4c gives another example where some of the cluttering square dots have overlapped in the bottom-right corner, resulting in a very large connected component whose size is even larger than that of the diamond-shaped dots - shown in Fig. 4d. The proposed breaking algorithm does not work successfully in the case demonstrated by the second example. The overlapping square dots would be incorrectly considered to be a diamond-shaped dot by the proposed breaking algorithm. To explore the performance of the erosion-based breaking algorithm, another 20 instances of Drawing CAPTCHAs were generated and attacked. Each instance contained 3 diamonds and had the form shown in the samples above. The correct rate of identification of diamonds was 91% (55 of 60). 5 diamonds were not identified correctly; consequently we can see that 15 of 20 sample CAPTCHAs were completely broken. This represents a 75% breaking rate. Breaking rates of even 10% make CAPTCHAs ineffective against automated attacks; consequently this attack means that the Drawing CAPTCHA can now be considered unsuitable for future real world use.

4 Proposed Image-based CAPTCHA

One of advantages humans possess over computers is the capacity to recognize similar objects within images, despite various types of distortion and transformation. We utilized this property of the human visual system in order to design a new type of CAPTCHA interface named "CAPTCHA Zoo", intended particularly for mobile devices and especially for younger users of mobile devices. It operates as follows.

A challenge image is created by first generating a background image textured with grass to obfuscate the image environment and make image-processing attacks more difficult. We chose this under the assumption that the human visual system will be well adapted to picking out shapes from naturally colored and arranged backgrounds, such as large areas of grass. Two visually similar kinds of animals (target animals, to be identified by the user, and noise/clutter animals, to obfuscate the challenge) are then randomly drawn over the background. The number of

Figure 5: Example of a CAPTCHA Zoo challenge.

target animals is much smaller than the number of noise/clutter animals. All animals are generated from 3D models with variation in color, lighting, and rotation. Further, animals may overlap one another as well as the textured background. This degree of obfuscation is important, because of known weaknesses of existing approaches.

For example, CAPTCHA techniques based on the use of 2-D photographic imagery of animals are vulnerable both to dictionary attacks (i.e. there may be a limited number of possible challenge images) and to attacks based upon invariants (e.g. a photograph of a cat's face might be distinguishable by the relative position and coloration of its eyes and nose). Equally, CAPTCHA techniques based upon presenting a rotated 2-D projection of a single 3-D animal model under fixed lighting and coloring, and without further visual clutter or obfuscation, might be vulnerable to attack either by noticing points of invariance between projections, or perhaps by reversing the 3-D rotation transformation and matching against a known pose.

Here, in the case where the attacker's problem is to recognize several parts of a varied arrangement of partly-overlaid projections of 3D models, under the conditions of varying color, lighting and rotation transforms (and potentially texture transforms), in the presence of visual clutter, we note a general solution does not presently exist. However, most humans can solve this task very conveniently and easily, as it is a natural skill present within the human visual system. Animals are objects which humans are naturally well-adapted to recognize under a variety of colorings, angles, and lighting conditions, and within complex, visually cluttered environments. An essentially infinite range of 2D projections, lightings and colorings can be generated here from a small number of models. The overlapping of the animals with each other and with a noisy background represents a further level of obfuscation that takes this visual recognition problem well beyond the capacity of demonstrated image-processing based CAPTCHA-breaking attacks seen to date. Users are asked to point out the locations of the target animals by pressing on the appropriate part of the touch screen. Fig. 5 gives a simple demonstrative example of CAPTCHA Zoo, using dogs and horses. Here, horses are the target animal, and dogs are the noise/clutter animal intended to confuse non-human users. Users pass the test by pointing out the locations of all three horses.

As it happens, this new form of CAPTCHA is very relevant to current themes in image-processing research. Visual object recognition has become a leading research topic within image processing. In fact, some advanced techniques have been proposed to recognize animal types in recent years, though they have focussed primarily on 2D images such as photographs. In 2007, Elson et. al introduced a technique with a success rate of about 83% in distinguishing cats and dogs (Elson et al. 2007). Although the recognition rate of that technique was high, it does not

Table 1: CAPTCHA Zoo usability testing results.

m	15	15	15	25	25	25	35	35	35
n	3	4	5	3	4	5	3	4	5
Passing rate	92%	90%	89%	94%	90%	91%	97%	96%	94%
Time (seconds)	0.45	0.72	0.80	0.48	0.60	0.80	0.49	0.59	0.76

Table 2: Drawing CAPTCHA and Mobile Text CAPTCHAs usability testing results.

	Drawing CAPTCHA	Badongo	Gimpy	MSN-type	Yahoo-type
Passing rate	84%	94%	94%	95%	95%
Time (seconds)	4.24	4.16	4.72	4.24	4.38

provide any leverage against this model. Breaking this newly proposed model is much more difficult; in Elson's work, every image only had one cat or one dog, whereas here, every image has multiple animals of different types, colors and orientations, which tend to overlap. Consequently it should be very difficult for an image-processing algorithm to isolate the target animals from CAPTCHA Zoo images, in practice.

An experimental analysis of the properties of this CAPTCHA was conducted using human subjects, as follows. Let the total number of animals and the number of target animals be denoted as m and n, respectively. The results of an experiment studying the effects of m and n is presented in Table 1. Three choices of m are shown: 15, 25, or 35; and three choices of n: 3, 4, or 5. The rate of correct responses by humans (passing rate), and the amount of time necessary for recognition are used to measure the performance of this new CAPTCHA type under varying parameters. A total of 30 humans took part in this experiment as subjects. CAPTCHA Zoo's performance is strongly affected by the choice and combination of m and n.

Three interesting phenomena result from this experiment. The first is that when the number of total animals is fixed, i.e., the value of m is fixed, a greater value of n results in a slightly reduced passing rate. In other words, an increase in the number of target animals has a very small but noticeable negative effect upon the human ability to pass the CAPTCHA. For example, when m=15, the passing rate is 92%, 90%, and 89% when the value of n is 3, 4, and 5, respectively. There are similar results when m is 25 or 35. A much more noticeable second phenomenon is the effect upon time taken to pass the CAPTCHA with n varying and m held constant; we see that the time taken to pass the CAPTCHA rises dramatically as n rises and the number of target animals to be selected increases. However is not clear whether this is due solely to the added time taken to enter the locations of target animals, or whether it is due also to an increased difficulty of recognition; further analysis would be needed to clarify this issue and unfortunately this would require a new study in this case. The third phenomenon discovered is that under the constraint of a fixed value of n, i.e., when the number of target animals is fixed, the passing rate does not decrease and the time taken does not increase as the value of m increases. Notice that rising m and fixed n means an increase only in the number of noise/clutter animals. Consequently, we believe that the human visual system is not perturbed by a modestly increasing level of noise in this problem, under the tested conditions. This is an interesting phenomenon, as in general, raising the amount of noise/clutter usually has a very negative impact upon the usability of traditional CAPTCHAs because of the sensitivity of text characters to disruption by noise. Further study would be needed to determine the limits of this phenomenon.

Figure 6: An example of a simplified text-based CAPTCHA suitable for mobile devices.

5 Comparison with the Drawing CAPTCHA and text-based CAPTCHAs.

We made an experimental comparison between CAPTCHA Zoo, the Drawing CAPTCHA, and text-based CAPTCHA challenges. We had noticed that existing text CAPTCHA systems presented in a web browser were awkward for users to select, zoom and respond to. We therefore invented an appropriate text CAPTCHA interface for mobile devices, which we used in combination with the obfuscation styles of several famous text CAPTCHA systems to try to enable fair comparison between the different types of CAPTCHA.

This Mobile Text CAPTCHA system has the following format: a four character text CAPTCHA containing a randomly chosen restricted range of case-insensitive characters, with a reduced-alphabet on-screen keyboard integrated into the CAPTCHA itself to simplify user text entry (some onscreen keyboards would overlap the challenge image). The security of such a CAPTCHA is less than that of a traditional desktop PC text CAPTCHA because of the reduced length and alphabet. However, we feel the increased suitability of this interface for a mobile screen allows a fairer comparison of usability between text-based CAPTCHAs and CAPTCHA Zoo. An example challenge is shown in Fig. 6.

Our Mobile Text CAPTCHA interface was used with samples representing each of four well known text systems (Badongo, Gimpy, MSN, Yahoo). Five challenges and 55 users were tested for each CAPTCHA system. The Drawing CAPTCHA was also tested during the same experiment. Table 2 indicates the results.

The passing rates for CAPTCHA Zoo and the Mobile Text CAPTCHA are similar; but the time taken to pass a CAPTCHA Zoo challenge appears to be noticeably smaller. However, we expect there may be more effective ways to present text CAPTCHAs for mobile devices than the Mobile Text CAPTCHA

we suggest here. We also found that the Drawing CAPTCHA has longer average responses times than CAPTCHA Zoo and a lower passing rate.

6 Conclusions

This paper has made two main contributions. The first is the proposal of an erosion-based CAPTCHA-breaking algorithm that successfully attacks the Drawing CAPTCHA for mobile devices. The second is a new CAPTCHA system for mobile devices (and particularly for younger users of mobile devices) called CAPTCHA Zoo, which is based on the parameterized 2D projection of 3D models of natural animals onto a natural background. This second contribution represents a challenge for automated bot agents that we feel is presently insurmountable by CAPTCHA-breaking techniques. CAPTCHA Zoo has been shown experimentally to be convenient for humans even under varying parameters. These contributions improve the security of online systems that are accessed by mobile devices without having a negative impact upon the usability or accessibility of such systems. We hope the results of this research will discourage attacks on systems accessed by mobile devices, and will open some new avenues for research into CAPTCHA interfaces that are suitable for mobile devices.

References

Pope, C., Kaur, K., (2005): Is It Human or Computer? Defending E-Commerce with CAPTCHAs. IT Professional, vol. 7, no. 2, pp. 4349.

Mori, G., Malik, J., (2005): Recognizing Objects in Adversarial Clutter: Breaking a Visual CAPTCHA. In Proc. CVPR'03, pp. 134-141.

Moy, G., Jones, N., Harkless, C., Potter, R., (2004): Distortion Estimation Techniques in Solving Visual CAPTCHAs. In Proc. CVPR'04, vol. 2, pp. 23-28.

Chellapilla, K., Larson, K., Simard, P., Czerwinski, M., (2005): Computers Beat Humans at Single Character Recognition in Reading Based Human Interaction Proofs (HIPs). In Proc. CEAS2005.

Chellapilla, K., Simard, P., (2005): Using Machine Learning to Break Visual Human Interaction Proofs (HIPs). In L. K. Saul, Y. Weiss, and L. Bottou, editors, Advances in Neural Information Processing Systems 17, pp. 265272. MIT Press.

Coates, L., Baird, H. S., Fateman, R.J., (2001): Pessimal Print: A Reverse Turing Test. In Proc. IDCAR2001, pp. 1154-1158.

Hoque, M. E., Russomanno, D. J., Yeasin, M., (2006): 2D Captchas from 3D Models. In Proc. IEEE SoutheastCon, pp. 165-170.

Baird, H. S., Bentley, J. L., (2005): Implicit CAPTCHAs. In Proc. Document Recognition and Retrieval XII, pp. 191-196.

Misra, D., Gaj, K., (2006): Face Recognition CAPTCHAs. In Proc. AICT2006, pp. 122-127.

Huang, S.Y., Lee, Y.K., Bell, G., Ou, Z.h., (2008): An efficient segmentation algorithm for CAPTCHAs with line cluttering and character warping. Multimedia Tools and Applications.

Shirali-Shahreza, M., Shirali-Shahreza, S., (2006): Drawing CAPTCHA. In Proc. ITI2006, pp. 475-480.

Elson, J., Douceur, J., Howell, J., Saul, J., (2007): Asirra: a CAPTCHA that exploits interest-aligned manual image categorization. In Proc. ACM CCS 2007. Alexandria, Virginia, USA: ACM.